COMMANDE D'UNE MACHINE A TRICOTER

Coll Jimmy
jimmycoll2806@gmail.com

RÉSUMÉ

Lorsque le tricot devient le centre de son activité professionnelle, les aiguilles deviennent obsolètes et il est primordial d'utiliser une machine comme un véritable industriel. C'est le cas de Claire Williams, une artiste qui a frappé à la porte de l'I.S.I.B. (Institut Supérieur Industriel de Bruxelles) pour demander de l'aide. Elle souhaitait faciliter l'utilisation de sa machine à tricoter via une commande électronique ajoutée. Pour réaliser au mieux ses souhaits, le moteur et la machine ont été analysés afin d'en comprendre le fonctionnement exact et de ne pas l'abîmer lorsque des composants supplémentaires seraient ajoutés. C'est à partir de ce moment-là que différentes hypothèses ont pu être envisagées et que différents tests ont pu avoir lieu. C'est finalement à l'aide d'un Arduino Uno et de quelques composants électroniques qu'une solution fonctionnelle a été mise en place. Elle peut maintenant commander sa machine à l'aide d'un bouton et voir sur un écran la progression du nombre de rangs effectués.

Mots-clefs : Machine à tricoter, commande, électronique, Arduino Uno.

1. INTRODUCTION

Chaque année, les étudiants en 3ème bachelier du groupe génie électrique de l'I.S.I.B. doivent faire un bureau d'études sur un sujet proposé par les professeurs. Dans cet article, il sera question de la commande d'un moteur d'une machine à tricoter[1].

Ce projet fut encadré par M. Grobet mais demandé à la base par une artiste, Claire Williams. Elle désirait pouvoir ajouter un module électronique à son moteur afin de le commander de différentes manières.

L'objectif principal de ce projet était d'établir une interface entre l'homme et la machine dont on connaissait entièrement le fonctionnement. Cette interface pourrait alors être modifiée et améliorée pour ajouter des fonctionnalités à la machine.

A la demande de C. Williams, les plans de cette interface devront être consultables par n'importe qui sur internet. Elle tient en effet à ce que ce travail soit open source.

2. ETUDE DE LA MACHINE

La première étape d'un tel travail consiste à étudier la machine et le moteur qui doivent être commandés. Dans le cas présent, il s'est avéré que la carte électronique était difficile d'accès et qu'aucune information n'était disponible sur internet. C'est grâce aux options disponibles sur le moteur original qu'une possibilité de commande a été trouvée.

2.1. Remplacement de la pédale de mise en marche

Le moteur acheté possédait une pédale permettant une mise en marche du moteur par une simple pression. Après démontage de celle-ci, on peut voir qu'il s'agit simplement d'établir une connexion pendant un temps court entre deux fils. C'est à ce moment-là que la solution à mettre en place fut trouvée. Il fallait désigner un circuit électronique qui ferait office d'interrupteur entre l'homme et le moteur.

3. PARTIE ELECTRONIQUE

3.1. Arduino Uno

La première étape pour mettre en place la solution était de choisir entre l'utilisation de l'Arduino Uno[2] et du Raspberry Pi[3]. Comme les caractéristiques de ces deux modules sont assez semblables, le choix s'est basé sur le langage de programmation. L'Arduino qui se programme dans un langage proche du Java et du C++, que je connaissais déjà, fut privilégié par rapport au Raspberry.

3.2. Circuit électronique

Pour établir un circuit électronique adéquat, il fallait réfléchir aux différentes contraintes auxquelles je pourrais être confronté. Il fallait pouvoir contrôler le temps d'impulsion indépendamment du temps d'appui sur le bouton. Je devais prêter attention au fait qu'un trop grand courant du moteur ne vienne pas griller les éléments du circuit. Un écran LCD (Liquid Crystal Display) a également été mis en place dès la première solution en prévision des améliorations futures. Pour répondre à ces différents critères, j'ai donc agencé un

[1] http://machineknittingetc.com/brother-kh910-service-manual.html

[2] https://www.arduino.cc/en/Main/arduinoBoardUno

[3] https://www.raspberrypi.org/

bouton, un transistor, une diode, l'écran LCD, un potentiomètre et quelques résistances.

Dans un deuxième temps, j'ai ajouté une fonction permettant de compter le nombre de rangs tricotés par la machine en temps réel. Pour cela j'ai récupéré la valeur de la tension aux bornes de deux LED présentes sur la machine (ces dernières étaient présentes pour indiquer le sens de fonctionnement du moteur).

3.2.1. Le bouton, le transistor et la diode

Ces trois éléments sont utilisés ensemble pour donner l'impulsion au démarrage du moteur. Le bouton sert à recevoir la commande de l'utilisateur. Le transistor permet de fermer le circuit pendant un certain temps et de lancer le moteur alors que la diode protège le transistor d'un éventuel retour.

3.2.2. L'écran LCD et le potentiomètre

Le potentiomètre est présent pour régler la luminosité de l'écran. L'écran LCD est celui fourni par le starter kit[4] d'Arduino et son utilisation est similaire à celle du projet 11 proposé dans le livre [1] fourni avec le kit.

3.2.3. Schéma électronique

Le schéma électronique (figure 1) met en place les éléments précités. Il a été réalisé à l'aide du logiciel libre fritzing[5].

Sur ce schéma :

- R1 est le potentiomètre
- R2, R3 et R4 sont des résistances
- S1 est le bouton (switch)
- Q1 est le transistor

Pour utiliser ce circuit, il faut brancher la masse du moteur au pin 3 du transistor et sa source au pin 2. Il faut également que l'Arduino soit chargé du code adéquat (voir partie informatique).

A ces éléments, il faut aussi ajouter une connexion en A0 et A1 des LED qui permettent le comptage des rangs (voir ANNEXE 1 – Mode d'emploi).

4. PARTIE INFORMATIQUE

Le code complet est disponible dans l'annexe 2. Cette section permettra de le comprendre afin de pouvoir être modifié par celui qui désire l'utiliser.

[4] https://www.arduino.cc/en/Main/ArduinoStarterKit

[5] http://fritzing.org/home/

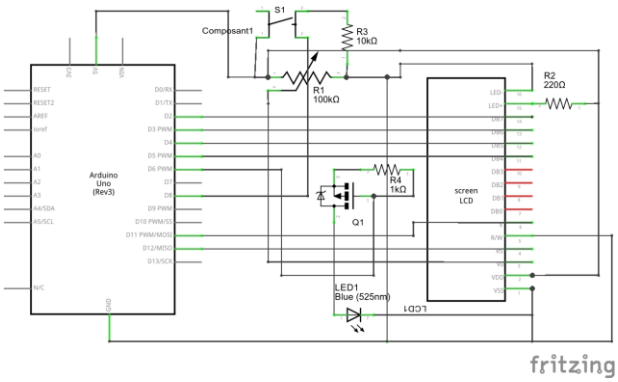

Figure 1. Schéma électronique

4.1. Liquid Crystal Display

Comme expliqué plus tôt, tout ce qui concerne cette partie provient d'un livre. Il ne faut pas toucher aux paramètres d'initialisation et pour l'utilisation, les commentaires dans le code suffisent pour comprendre leur fonctionnement.

4.2. Constantes et paramètres

Les constantes définissent les pins de l'Arduino auxquels sont branchés les différents éléments. Le « motorPin » correspond au pin du transistor permettant de lancer ou d'arrêter le moteur quand il change d'état.

Les paramètres sont tous définis pour une utilisation correcte de la machine. Le paramètre « numberRow » peut être modifié par l'utilisateur en fonction du nombre de rangs qu'il veut faire.

4.3. Le « void setup »

Cette fonction fait partie des deux fonctions indispensables au bon fonctionnement d'un Arduino. Le programme ne passe qu'une fois dans cette fonction au moment de l'exécution. Ici le compteur se remet à zéro et l'écran affiche le nombre de rangs qui sont programmés. Tant que l'Arduino n'est pas remis à zéro (reset), le compteur ne revient pas à zéro.

4.4. Le « void loop »

C'est la deuxième fonction indispensable à l'Arduino. Il exécute en continu la boucle tant que l'Arduino est sous tension. On voit ici qu'il regarde l'état du bouton et qu'il change l'état du transistor lorsque le bouton est pressé.

Les deux fonctions suivantes permettent de compter le nombre de rangs effectués. Lorsqu'une LED s'allume, le programme constate le changement de tension aux bornes de celle-ci et comptabilise un rang. Quand le compteur atteint le nombre de rangs programmés par l'utilisateur le moteur s'arrête.

5. RESULTATS FINAUX

C. Williams peut à présent utiliser son moteur à partir d'un Arduino Uno. Elle peut choisir le nombre de rangs que la machine doit faire en modifiant une seule variable dans le programme fourni.

En cas de problème pendant le tricot, elle peut arrêter le moteur par une simple pression sur le bouton. Le compteur ne se remet pas à zéro et elle a donc la possibilité de redémarrer le moteur quand le problème est réglé.

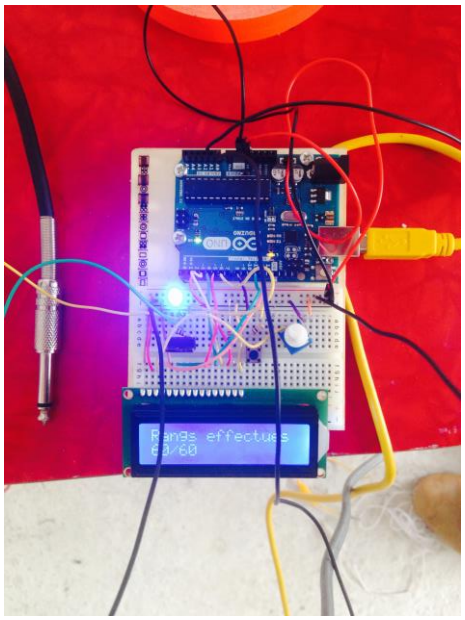

Figure 2 – Fin de tricot

Quand le tricot est fini, l'écran affiche le nombre de rangs effectués (Figure 2) et la machine s'arrête. Si l'utilisateur appuie sur le bouton il ne se passe rien, il doit réinitialiser l'Arduino pour refaire 60 rangs. S'il désire modifier le nombre de rangs, il doit repasser par l'ordinateur.

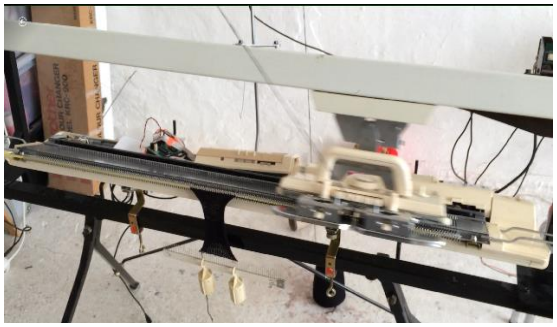

Figure 3- Machine à tricoter en fonction

Pour les personnes qui ne connaissaient pas l'existence des machines à tricoter, le tricot (figure 3) s'agrandit d'un rang à chaque passage du rail par-dessus le tricot. Habituellement le passage se fait manuellement mais c'est un travail qui demande beaucoup d'énergie. C'est pourquoi le moteur se greffe au-dessus de la partie qu'un utilisateur devrait attraper et effectuer les allers-retours à sa place.

6. CONCLUSIONS ET AMELIORATIONS POSSIBLES

Avant de conclure, je vais passer en revue les différentes possibilités d'amélioration qui pourraient encore être apportées au projet.

Tout d'abord, il faudrait faire un circuit imprimé pour éviter d'avoir des fils dans tous les sens et ne pas risquer de faux contacts ou des problèmes de connexion. Ce circuit n'est pas encore fait car quelques fonctionnalités basiques peuvent encore être ajoutées avant de le finaliser.

C. Williams souhaiterait pouvoir travailler avec un détecteur de mouvement à la place de l'interrupteur pour des foires d'exposition. La machine se mettrait en marche lorsqu'un spectateur passerait devant. Ça ne semble pas trop compliqué à mettre en place, il faudra prendre un peu de temps pour effectuer des tests. J'ai aussi pensé à un allumage au son d'un claquement de main afin de pouvoir l'allumer à distance ou encore avec une télécommande infrarouge par exemple.

Un dernier élément auquel j'ai pensé, mais qui nécessite un nouveau travail de recherches, serait l'implémentation d'un clavier numérique complet. Il permettrait de ne plus devoir passer par l'ordinateur pour programmer le nombre de rangs et rendrait le système de contrôle complètement indépendant.

Pour conclure, je pense que le travail effectué jusqu'ici répond au cahier des charges initialement prévu. Il est maintenant possible de commander le moteur et de programmer le nombre de rangs à effectuer. Le bouton sert d'arrêt d'urgence et la carte électronique de base du moteur n'a pas été perturbé par le montage, il a donc conservé toutes ses fonctionnalités de base.

7. REFERENCES

[1] Fitzgerald S. and Shiloh M., **Arduino Projects Book**, Turin, Italie, Septembre 2012